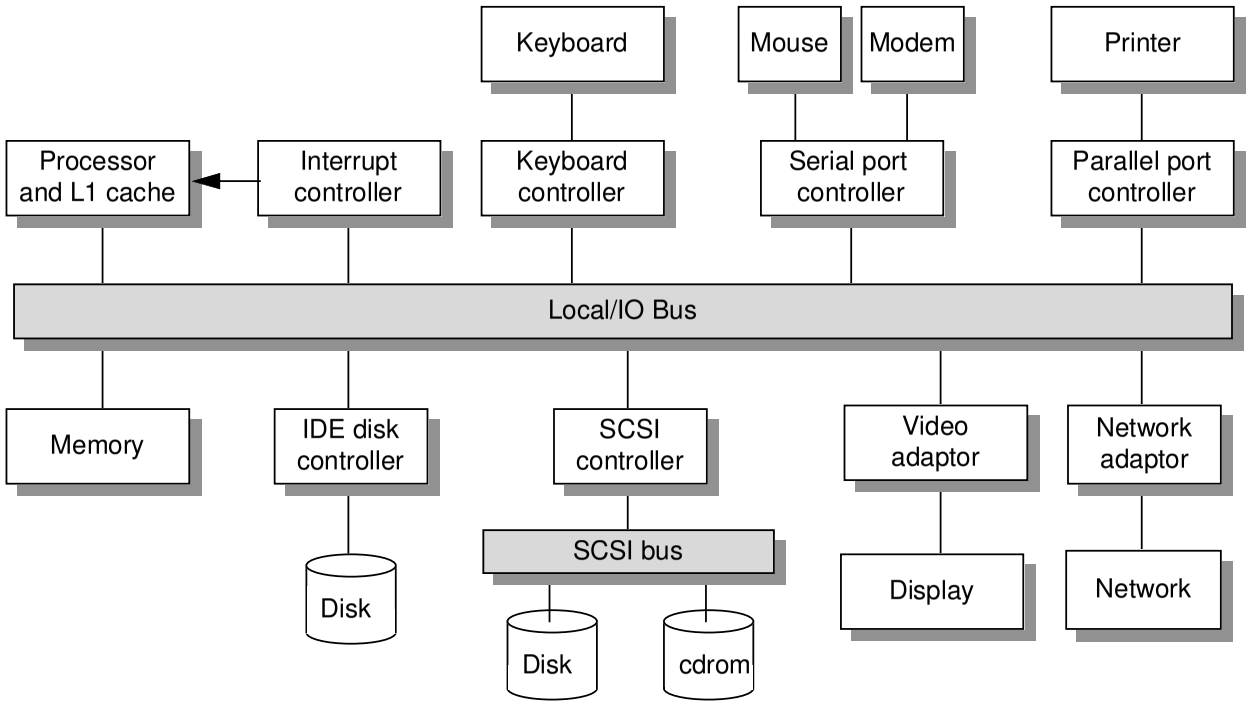
I/O Devices

April 17, 1996

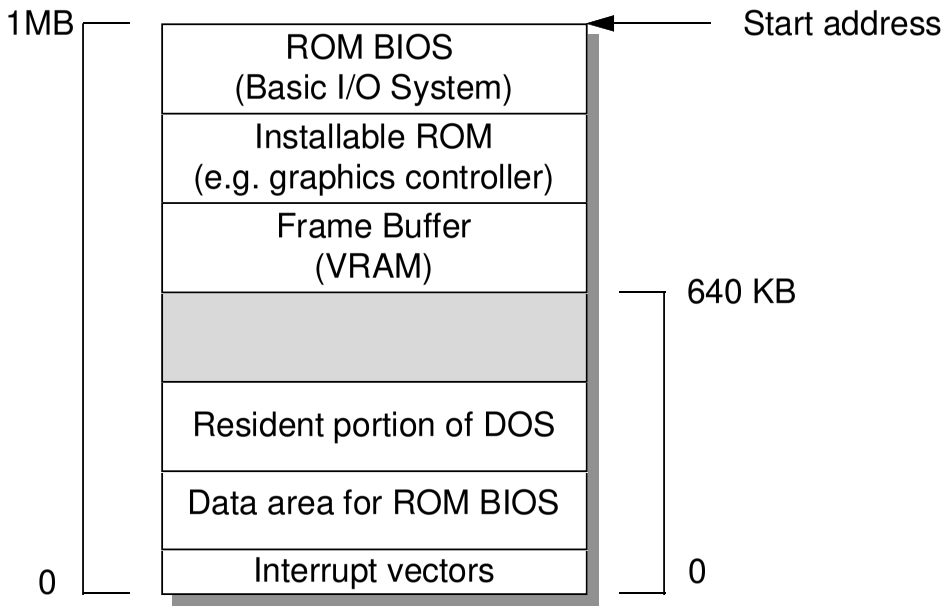
Topics:

- **PC memory and I/O spaces**
- **PC interrupts**
- **serial and parallel ports**
- **keyboard, mouse, display, and disks**
- **disks and capital murder**

PC organization



PC system address space



Boot process

- **Jump to start address in ROM BIOS**
- **Setup interrupt vector table**
- **Initialize standard chips and standard controllers**
  - **E.g., interrupt, display, serial parallel, and keyboard controllers**
- **Initialize optional devices**
  - **E.g., SCSI devices**
  - **config info contained in 64-byte battery backed "CMOS memory"**
- **Read bootstrap loader from sector 0 (the boot sector) of default disk**
- **Load resident portion of OS**
- **Pass control to OS**

PC I/O space

Separate I/O space with 64K addresses (ports)

- each device assigned one or more ports (for control & data)
- bus control line distinguishes I/O address from regular addresses

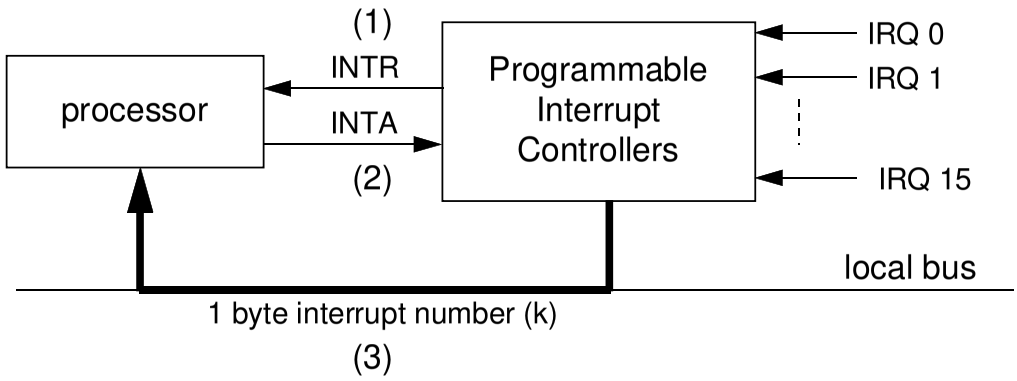
Separate instructions read and write to ports

- in $1, 0x60 -- read contents of keyboard data register

Some example ports:

0x60-0x6f	Keyboard controller
0x1f0-0x1f8	Hard disk controller
0x3bc-0x3f	Parallel port controller
0x3c0-0x3cf	Video adapter
0x3f8-0x3ff	Serial port controller

PC interrupts



(4) Take exception to address k*4

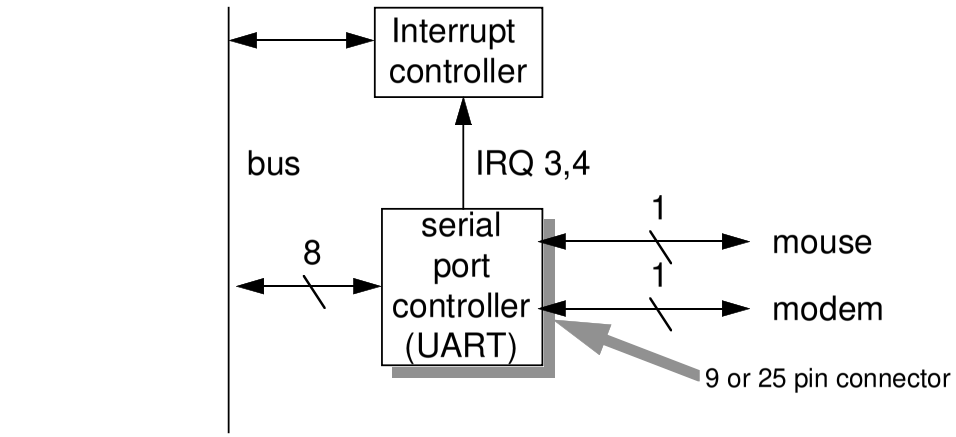
Example PC interrupts

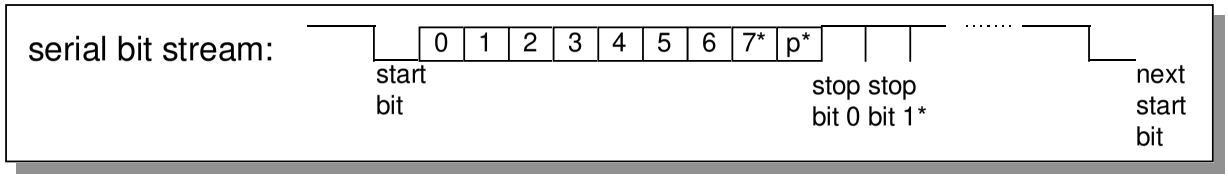
Interrupt #	Use	Type
0x00	divide by zero	internal
0x01	single-step	internal
0x03	breakpoint	internal
0x04	integer overflow	internal
0x08	IRQ 0 (timer tick)	external
0x09	IRQ 1 (keyboard)	external
0x0a	IRQ 2	external
0x21	system call to BIOS	internal
0x1c	timer tick	external

highest priority

lowest priority

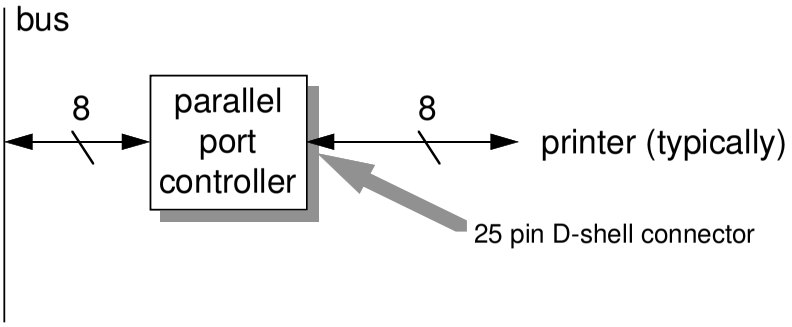
Serial ports





slow (10-20 Kbits/sec transfer rate) but long and flexible

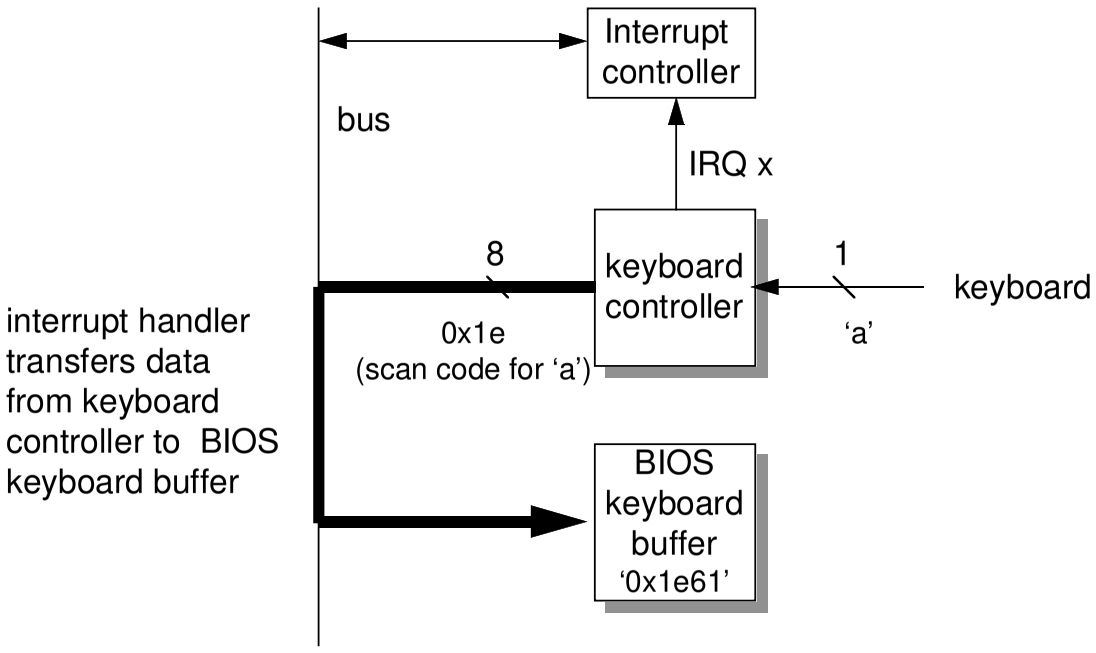
Parallel ports



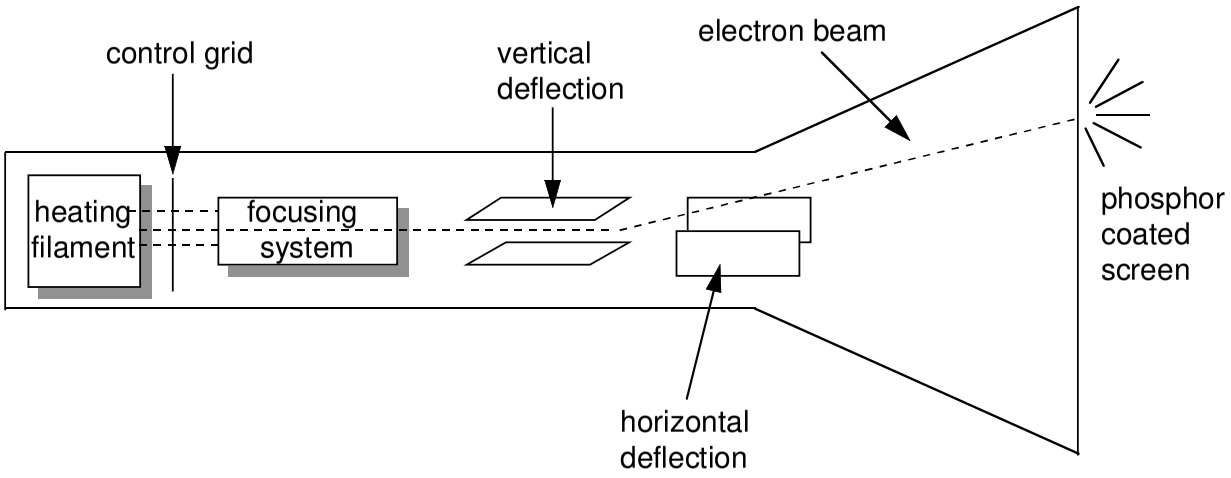
Each parallel port has three I/O ports:	
0x3bc	data (read/write)
0x3bd	status (read only)
0x3be	control (poll or interrupt)

faster, but shorter than serial ports
programmed, rather than interrupt driven I/O

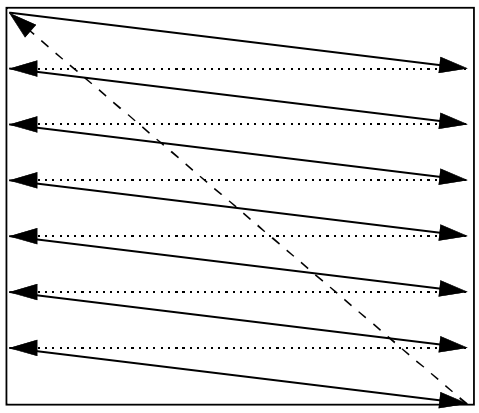
Keyboard



Display

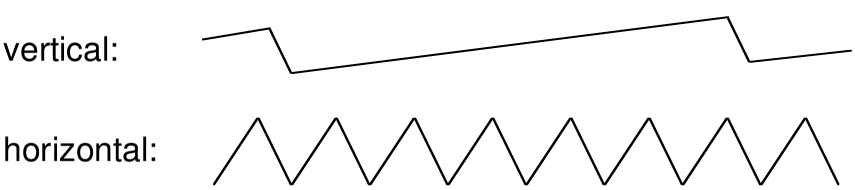


Raster scan

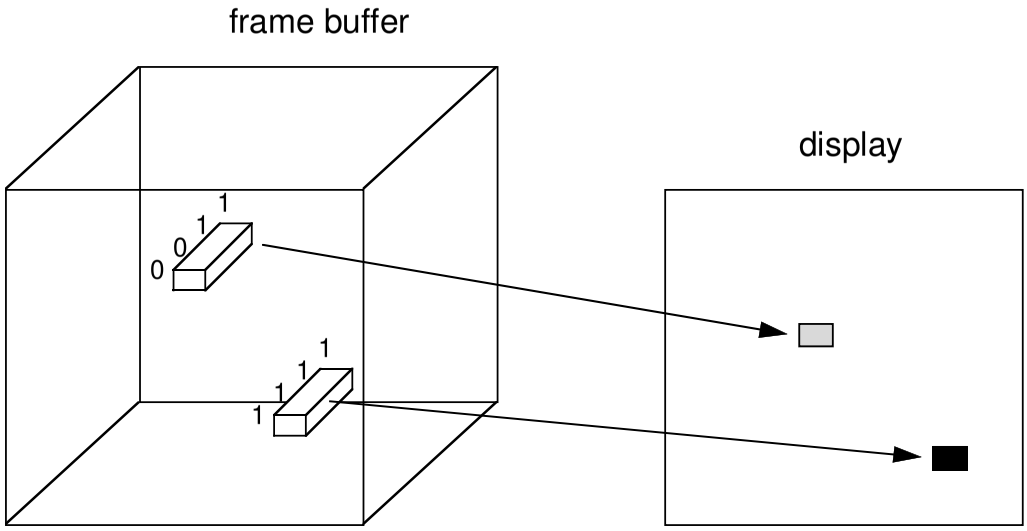


horizontal retrace

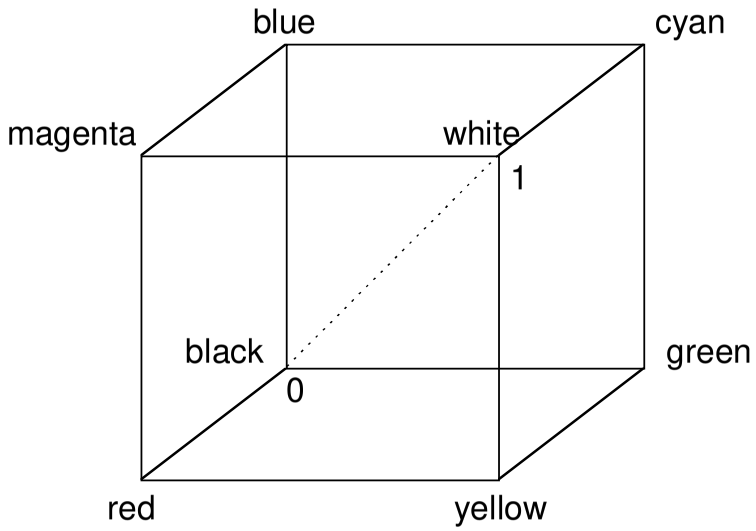
vertical retrace



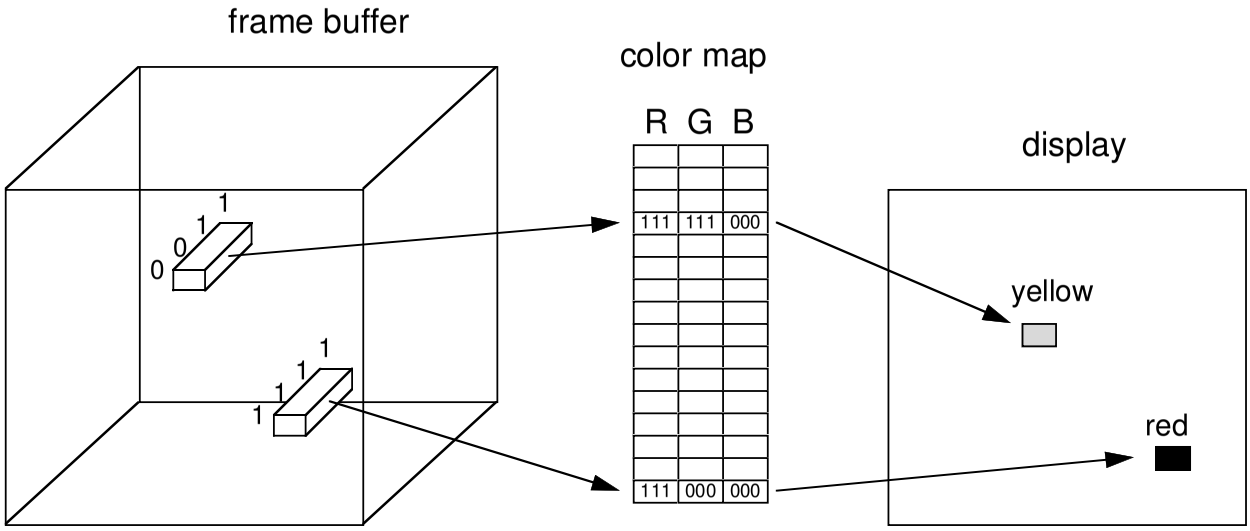
Frame buffer (grayscale)



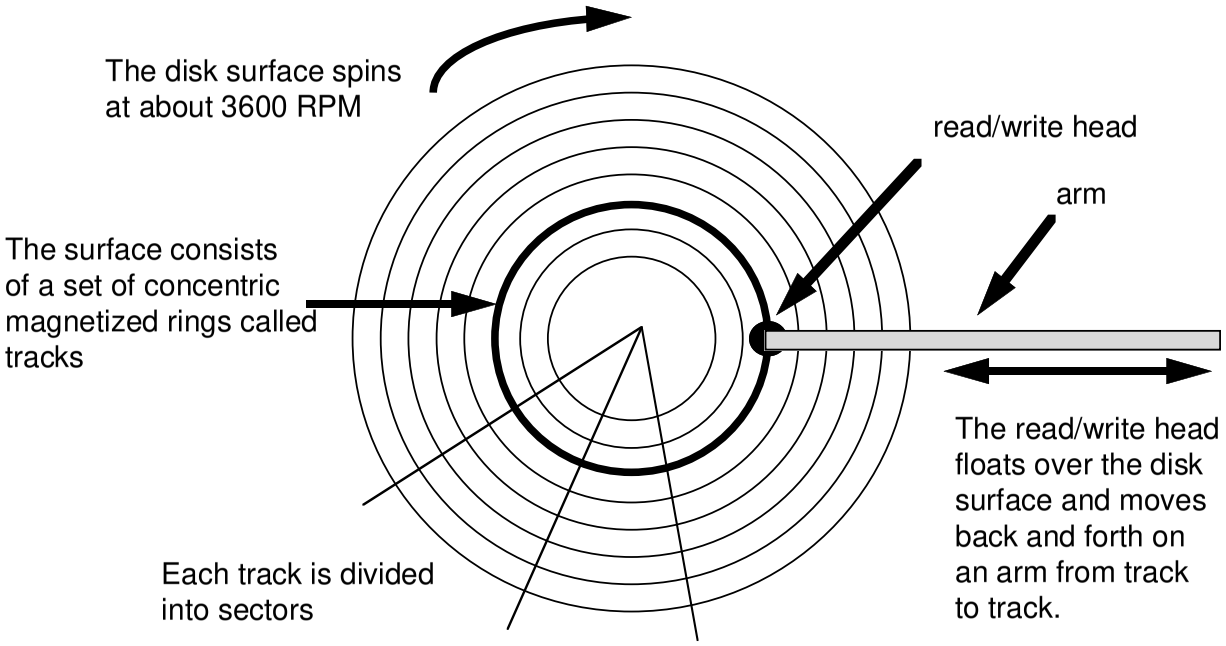
The RGB color space



Frame buffer (color)



Disks



Case Study: Copenhefer's Blunder

The DOS file system and its role
in a capital murder case

David O'Hallaron

Copenhefer Case

June 17, 1988 (Erie, PA)

- **Sally Weiner, wife of bank executive Harry Weiner, is kidnapped, held for ransom, and then murdered before the money can be delivered.**

June 27, 1988 (Erie, PA)

- **State trooper notices computer-generated sign in the window of a bookstore owned by David Copenhefer that looks similar to the ransom note. Becomes the basis for a search warrant.**
- **Police obtain warrant, and the FBI finds deleted versions of the ransom note and the murder plan on the disk drives in the PC's in the bookstore and Copenhefer's house.**

May, 1989 (Pittsburgh, PA)

- **Copenhefer sentenced to die.**
- **Still in the appeals process.**

How did he get caught?

He didn't understand the PC's DOS filesystem.

- The data in a deleted file is still on the disk!

The FBI knew this and searched the tracks of the disk for the character string "exactely", a misspelling that appears several times in the ransom note.

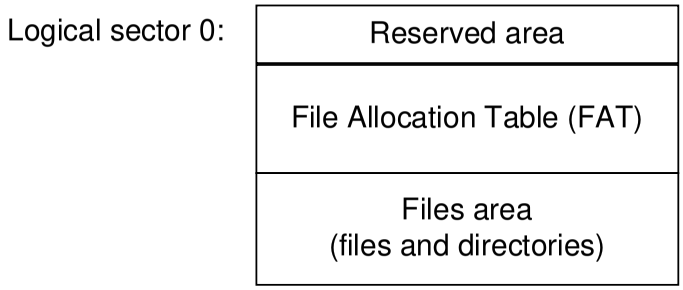
In 1994, I examined both of Copenhefer's computers as an expert witness to the Commonwealth of PA , "undeleted" the ransom note, and printed it out.

DOS File System

The disk is treated as a linear sequence of n "logical sectors", each 512 bytes in length:

- sector 0, sector 1, sector 2, ..., sector n-2, sector n-1

DOS Disk Map



Directory entries

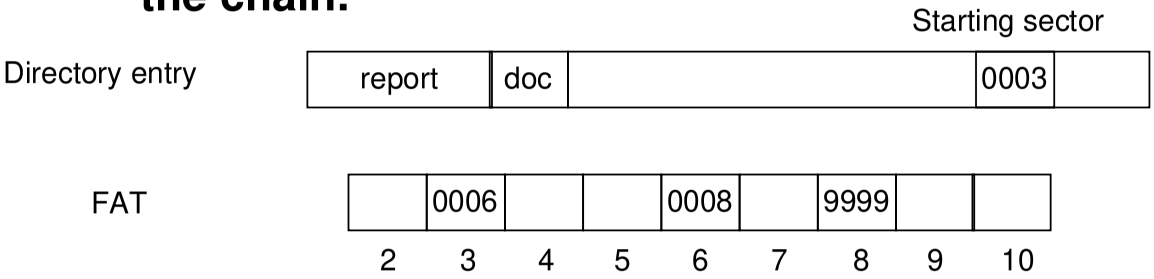
The eight parts of a directory entry

- **filename (8 bytes) e.g., report.doc**
- **filename extension (3 bytes) e.g., report.doc**
- **attribute (1 byte) e.g., file or directory, read only or read/write**
- **unused (10 bytes)**
- **time (2 bytes)**
- **date (2 bytes)**
- **starting sector number (2 bytes)**
- **file size (4 bytes)**

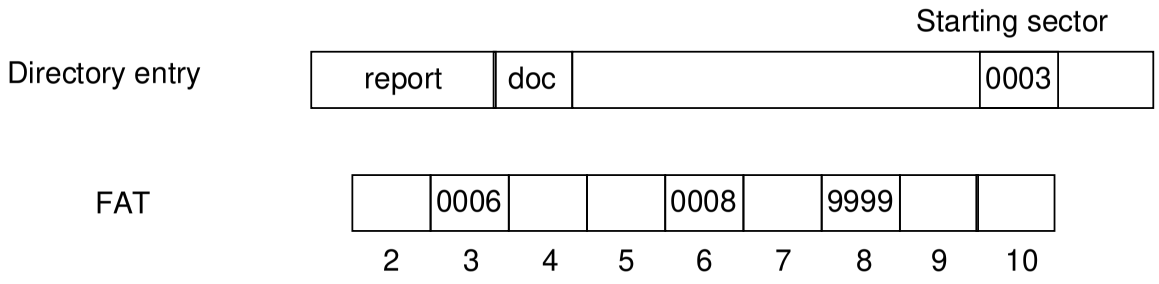
File Allocation Table (FAT)

The FAT is a sequence of 16 bit entries. The ith FAT entry corresponds to the ith logical disk sector.

The values of the entries form a chain that shows which logical sectors contain the data in a file or directory entry. “9999” ends the chain.

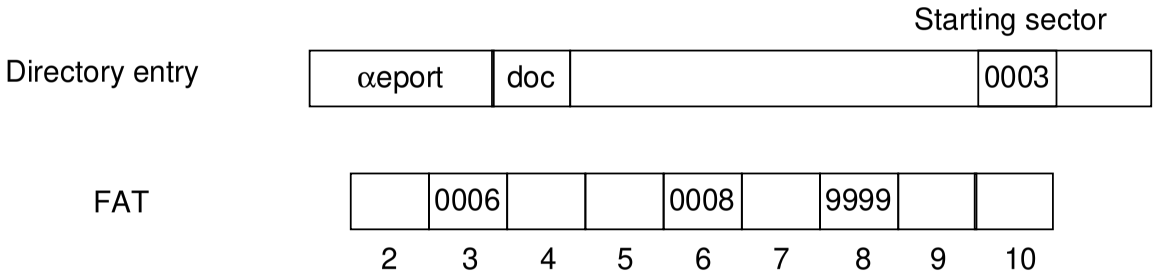


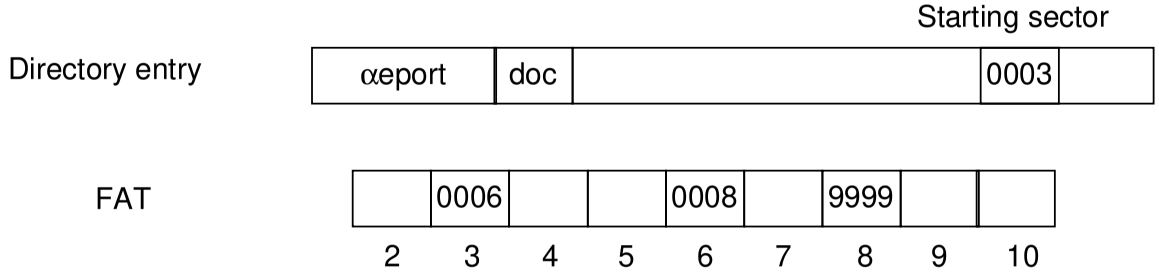
Example FAT chain



Deleting a file

When a file is deleted, the first word in the directory is changed to a special character (we'll call it 'α'). The FAT chain and the data are intact:



Recovering a deleted file

Very simple: look for all occurances of 'α' and follow the FAT chain to recreate the file. This is what I did (and the FBI could have done) with Copenhefer's ransom note. The only lost information is the first letterof the file name.